

CLAIMS

What is claimed is:

1. A method of selecting a default locality for a memory object requested by  
5 a process running on a central processing unit (CPU) in a multiprocessor system, the method comprising:  
determining whether the memory object comprises a shared-memory object;  
selecting the default locality to be within interleaved memory of the  
10 system if the memory object comprises said shared-memory object.
2. The method of claim 1, further comprising:  
determining whether the memory object comprises a stack-type object;  
15 and  
if the memory object comprises the stack-type object, then selecting the default locality to be within local memory at a same cell as the requesting CPU.
- 20 3. The method of claim 2, further comprising:  
if the memory object comprises a non-stack-type object, then determining whether the requesting process has threads running on multiple cells.
- 25 4. The method of claim 3, further comprising:  
if the requesting process has threads running on said multiple cells, then selecting the default locality to be within the interleaved memory of the system.
- 30 5. The method of claim 3, further comprising:  
if the requesting process has threads running on only one cell, then selecting the default locality to be within the local memory at the same cell as the requesting CPU.

6. The method of claim 1, wherein access time to the interleaved memory is approximately equal for each CPU in the multiprocessor system.
- 5 7. The method of claim 2, wherein the stack-type object comprises a process stack.
8. The method of claim 2, wherein the stack-type object comprises a UAREA type object.
- 10 9. A multiprocessor computing system, the system comprising:  
multiple symmetric multiprocessing (SMP) nodes;  
multiple central processing units (CPUs) at each SMP node;  
a memory control unit at each SMP node which is coupled to each CPU at  
15 that SMP node;  
shared memory at each SMP node which is accessible by way of the  
memory control unit at that SMP node;  
a switching system coupled to the memory control units so as to  
interconnect the multiple SMP nodes;  
20 an operating system running on the CPUs; and  
a virtual memory (VM) fault handler within the operating system,  
wherein the shared memory includes both local memory and interleaved  
memory, and  
wherein the VM fault handler executes instructions to provide intelligent  
25 default locality selection for memory objects requested by a  
process running on a CPU.
- 30 10. The multiprocessor computing system of claim 9, wherein said instructions executed by the VM fault handler (a) determine whether the memory object comprises a shared-memory object, and (b) select the default locality to be within the interleaved memory of the system if the memory object comprises said shared-memory object.

11. The multiprocessor computing system of claim 10, wherein said instructions executed by the VM fault handler also (c) determine whether the memory object comprises a stack-type object, and (d) select the default locality to be within local memory at a same SMP node as the requesting CPU if the memory object comprises the stack-type object.
12. The multiprocessor computing system of claim 11, wherein said instructions executed by the VM fault handler also (e) determine whether the requesting process has threads running on multiple SMP nodes if the memory object comprises a non-stack-type object, (f) select the default locality to be within the interleaved memory of the system if the requesting process has threads running on said multiple SMP nodes, and (g) select the default locality to be within the local memory at the same SMP node as the requesting CPU if the requesting process has threads running on only one SMP node.
13. The multiprocessor computing system of claim 9, wherein the switching system includes multiple switches interconnected together.
14. The multiprocessor computing system of claim 9, wherein the operating system comprises a type of UNIX operating system, and wherein the operating system supports multi-threaded processes.
15. The multiprocessor computing system of claim 9, wherein the operating system comprises a type of Windows operating system, and wherein the operating system supports multi-threaded processes.
16. The multiprocessor computing system of claim 9, wherein access time to the interleaved memory is approximately equal for each CPU in the multiprocessor system.
17. The multiprocessor computing system of claim 11, wherein the stack-type object comprises a process stack.

18. The multiprocessor computing system of claim 11, wherein the stack-type object comprises a UAREA type object.
- 5 19. An operating system for a multiprocessor computing system, the operating system comprising:  
a virtual memory manager configured for extending a memory space  
beyond limits of a physical address space; and  
a page fault handler to interrupt execution of the virtual memory manager  
10 when a page fault occurs,  
wherein the page fault handler is configured to provide intelligent default  
locality selection for memory objects.
- 15 20. The operating system of claim 19, wherein the intelligent default locality selection is configured to choose from a set of choices including local memories residing at each node of the system and interleaved memory which is striped to reside on multiple nodes of the system.
- 20